Homework 3

swarts2

February 2016

1 4.2

n a population, the correlation coefficient between family income and child IQ is 0.30. The mean family income was \$60, 000. The standard deviation in income is \$20, 000. IQ is measured on a scale such that the mean is 100, and the standard deviation is 15.

(a) Using this information, predict the expected IQ of a child whose family income is \$70, 000
(b) How reliable do you expect this prediction to be? Why? (your answer should be a property of correlation, not an opinion about IQ)
(c) The family income now rises does the correlation predict that the child will have a higher IQ? Why?

1.1 A

This equation was derived part way in the text, so I flushed it out fully. It predicts values based on simple linear regression.

$$Y^p = r * \frac{SD_y}{SD_x}(X) + (\mu_y - r * \frac{SD_y}{SD_x} * \mu_x)$$

$$Y^p = .3 * \frac{15}{20,000}(70,000) + 100 - .3\frac{15}{20,000}60,000$$

$$Y^p = 102.25$$

1.2 B

For this question, I will use the fact derived in the book that the standard deviation of the ERRORS term is $\sqrt{1-r^2}$. Also the book says that that small correlations have high probabilities of error. In this case $\sqrt{1-0.3^2} = .9539$ This seems like a rather large standard error, so I am going to say the prediction is unreliable.

1.3 C

Unless this is a trick question about the prediction being unreliable, (in which case question C is mute) the equation presented in part A clearly shows the answer is yes, higher income leads to higher IQ. Also there is the principle that a positive correlation value indicates that large values of X predict large values of Y

2 4.7

I did the programming exercise about the earth temperature below. It is straightforward to build a dataset (T,nt) where each entry contains the temperature of the earth (T) and the number of counties where FEMA de- clared tornadoes nt (for each year, you look up T and nt, and make a data item). I computed: mean (T) = 0.175, std (T) = 0.231, mean (nt) = 31.6, std (nt) = 30.8, and corr (T)nt = 0.471. What is the best prediction using this information for the number of tornadoes if the global earth temperature is 0.5? 0.6? 0.7?

2.1 0.5, 0.6, and 0.7

For this problem, I will once again use the formula I used in 4.2 C. I will also round to the nearest disaster.

$$Y^p = r * \frac{SD_y}{SD_x}(X) + (\mu_y - r * \frac{SD_y}{SD_x} * \mu_x)$$

$$Y^p_{.05} = .471 * (\frac{30.8}{.231}) * 0.5 - (\frac{30.8}{.231}) * .175 + 31.6$$

$$Y^p_{.06} = .471 * (\frac{30.8}{.231}) * 0.6 - (\frac{30.8}{.231}) * .175 + 31.6$$

$$Y^p_{.07} = .471 * (\frac{30.8}{.231}) * 0.7 - (\frac{30.8}{.231}) * .175 + 31.6$$

$$Y^P_(0.5) = 39.\overline{6} \approx 40$$

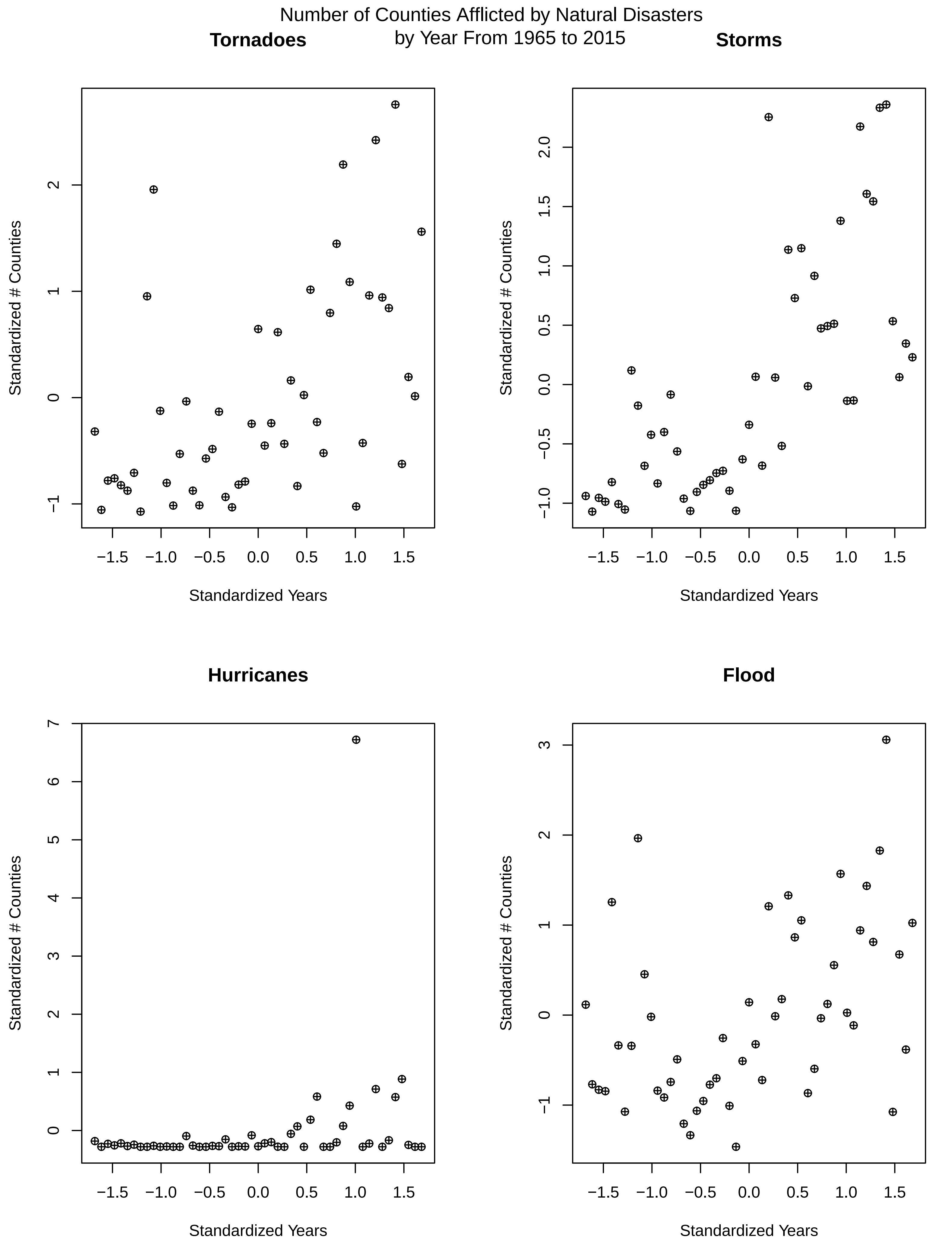
$$Y^P_(0.6) = 45.94\overline{6} \approx 46$$

$$Y^P_(0.7) = 52.22\overline{6} \approx 52$$

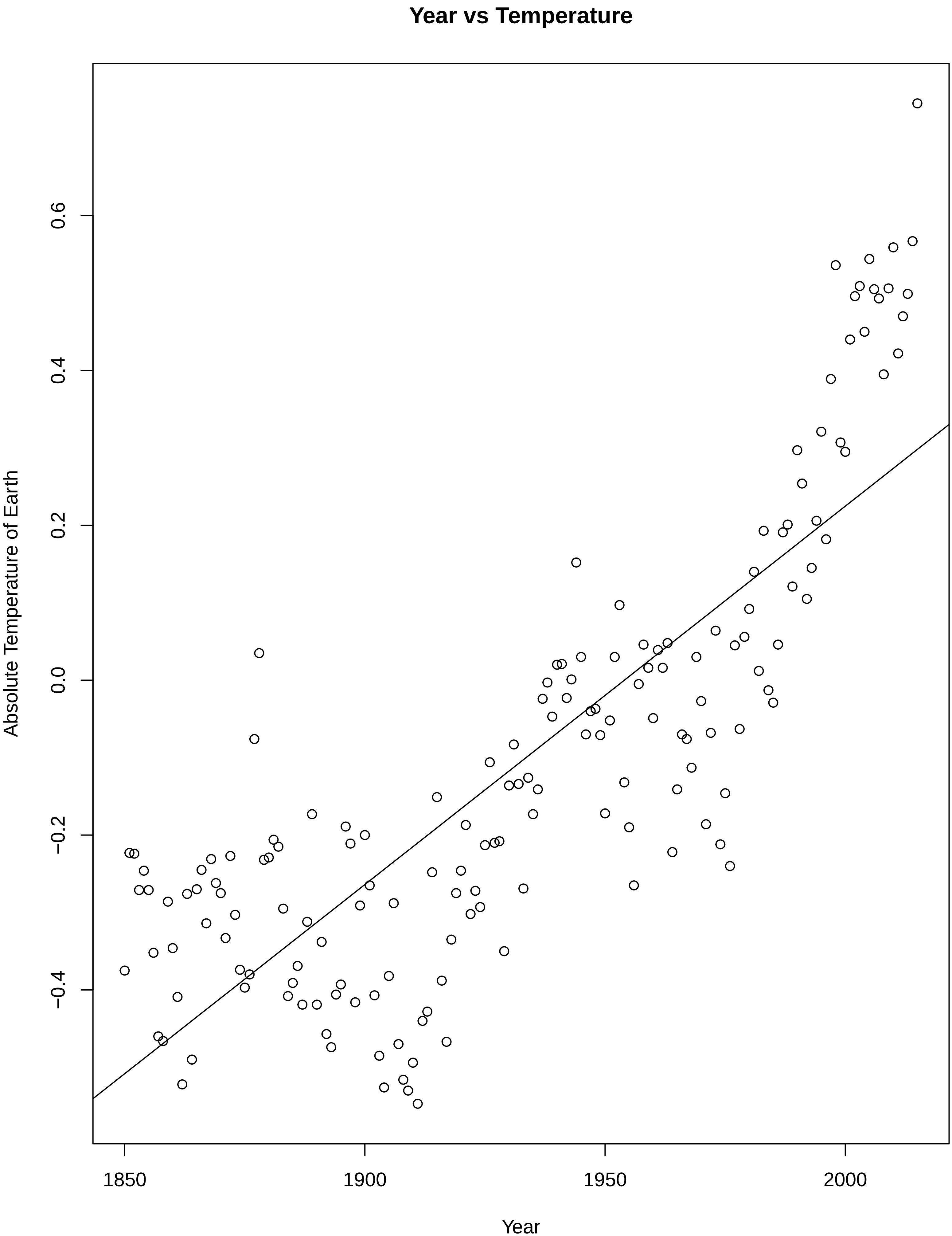
3 4.9

Below are the graphs I produced for the questions. and here is an index.

4.9 B
4.9 G



Number of Counties Afflicted by Natural Disasters
by Year From 1965 to 2015
Tornadoes
Storms
Hurricanes
Flood
Standardized # Counties
Standardized Years



Year vs Temperature
Absolute Temperature of Earth
Year
1850
1900
1950
2000
-0.4
-0.2
0.0
0.2
0.4
0.6

Here is the code for 4.9, and all the answers to the questions are in the code.

```
#Reading in the data
gtemp <- read.csv('~/Desktop/GlobalTemps.csv',header=TRUE)
gdist <- read.csv('~/Desktop/Disasters.csv')
head(gdist)
colnames(gdist)

#Making a numeric vector for each disaster type
#which keeps track of all the entries with
#titles with that kind of disaster in them.

storms <- NULL
storms <- rep(0,length(gdist$Title))
storms[grep('STORM',gdist$Title)] <- storms[grep('STORM',gdist$Title)]+1
storms

hurricanes <- NULL
hurricanes <- rep(0,length(gdist$Title))
hurricanes[grep('HURRICANE',gdist$Title)] <- hurricanes[
  grep('HURRICANE',gdist$Title)]+1
hurricanes

flooding <- NULL
flooding <- rep(0,length(gdist$Title))
flooding[grep('FLOODING',gdist$Title)] <- flooding[grep(
  'FLOODING',gdist$Title)]+1
flooding

tornade <- NULL
tornado <- rep(0,length(gdist$Title))
tornado[grep('TORNADO',gdist$Title)] <-tornado[grep(
  'TORNADO',gdist$Title)]+1
tornado

#Making a dataframe for those vectors.

weathers <- data.frame(storms,hurricanes,flooding,tornado)
weathers
#Pandas are a statistical term for data items that are to be thrown out.
#In this case, I am mainly throwing out drought events, because we don't
#really care about them for this assignment. And throwing them out doesn't
#affect anything.
pandas <- c(0)

#This loop goes through and reallocates points to each member based on whether
#a disaster of another type was also declared in that instance.

  for(i in 1:nrow(weathers)){
    if(sum(weathers[i,])==0){
      print("new panda")
```

```
      pandas <- c(pandas,i)
    }
    else{
      weathers[i,] <- weathers[i,]/sum(weathers[i,])
    }}
#Want to throw out the initial value that told R pandas is a numeric vector.
pandas <- pandas[1:length(pandas-1)]

#Add the date of each event
weathers<- (cbind(weathers,gdist$Declaration.Date))

#KILL THE PANDAS!!! lol jk, but they really are called pandas.
#It comes from the pandas dataset.
weathers <- weathers[-pandas[1:length(pandas)],]
head(weathers)

#as of this date, 2016 isn't over, so I am throwing those
#observations out too. The years we are working with are
#from 1953 to 2015
years <- c(1965:2015)
years

#A data frame to hold the number of disasters in that county in that year.
yearsVcounties <- weathers[1:length(years),1:4]
for(i in years)
{
  yearWeather <- weathers[grep(i,weathers$`gdist$Declaration.Date`),1:4]
  yearsVcounties[i-1964,] <- sapply(yearWeather[1:4],sum)
}

#4.9 A!, and I will be putting a copy in the PDF

yearsVcounties <- cbind(yearsVcounties,years)
yearsVcounties
write.table(yearsVcounties, file = "yearsVCounties.csv",
            row.names=FALSE, na='',col.names=TRUE, sep=",")

#These means and std's will be used in the formulas.

yearmean<- mean(years)
yearstd <- sd(years)

stormean <- mean(yearsVcounties$storms)
storstd <- sd(yearsVcounties$storms)

tornmean <- mean(yearsVcounties$tornado)
tornstd <- sd(yearsVcounties$tornado)

flodmean <- mean(yearsVcounties$flooding)
flodstd <- sd(yearsVcounties$flooding)

hurrmean <- mean(yearsVcounties$hurricanes)
```

```
hurrstd <- sd(yearsVcounties$hurricanes)

standardYears <- c(rep(1,length(years)))
standardStor <- c(rep(1,length(years)))
standardTorn <- c(rep(1,length(years)))
standardFlod <- c(rep(1,length(years)))
standardHurr <- c(rep(1,length(years)))

#This following loop creates the standardized
#versions of each column in yearsVcounties

for(i in 1:length(years))
{
  standardYears[i] <- (years[i]-yearmean)/yearstd
  standardStor[i] <- (yearsVcounties$storms[i]-stormean)/storstd
  standardTorn[i] <- (yearsVcounties$tornado[i]-tornmean)/tornstd
  standardFlod[i] <- (yearsVcounties$flooding[i]-flodmean)/flodstd
  standardHurr[i] <- (yearsVcounties$hurricanes[i]-hurrmean)/hurrstd
}
standardYears

#Binding together the raw and standardized data.
yearsVcounties <- cbind(yearsVcounties,
    standardYears,standardStor,standardTorn,standardFlod,standardHurr)
yearsVcounties


#4.9 B!
#I would like to point out that at this level of statistics, normalized and standardized
# are interchangeable
par(mfrow=c(2,2),mar=c(5.1,4.1,5.1,3.1))
plot(yearsVcounties$standardYears,yearsVcounties$standardTorn,
  xlab='Standardized Years',ylab='Standardized # Counties',
  main='Tornadoes',pch=10)
plot(yearsVcounties$standardYears,yearsVcounties$standardStor,
     xlab='Standardized Years',ylab='Standardized # Counties',
     main='Storms',pch=10)
plot(yearsVcounties$standardYears,yearsVcounties$standardHurr,
     xlab='Standardized Years',ylab='Standardized # Counties',
     main='Hurricanes',pch=10)
plot(yearsVcounties$standardYears,yearsVcounties$standardFlod,
     xlab='Standardized Years',ylab='Standardized # Counties',
     main='Flood',pch=10)
mtext("Number of Counties Afflicted by Natural Disasters
      by Year From 1965 to 2015", side = 3, line = -3, outer = TRUE)


dev.off()

plot(years[45:length(years)],yearsVcounties$storms[45:length(years)])
plot(years,yearsVcounties$tornado)
tail(yearsVcounties$hurricanes,11)
```

```
#Correlation coefficients for each disaster.
rstor <- cor(yearsVcounties$standardYears,yearsVcounties$standardStor)
rhurr <- cor(yearsVcounties$standardYears,yearsVcounties$standardHurr)
rtorn <- cor(yearsVcounties$standardYears,yearsVcounties$standardTorn)
rflod <- cor(yearsVcounties$standardYears,yearsVcounties$standardFlod)

#Again the formula is r(std(Y)/std(X))(x-mean(x))+mean(y)

#4.9 C!

rstor*(storstd/yearstd)*(2013-yearmean)+stormean
yearsVcounties$storm[grep('2013',years)]
#predicted number of storms for 2013 is 635.8441 or 608 ish
#actual number of storms is 341.3333 or 341ish
#This prediction is off by roughly 80 percent if this was a physics class
#But actually because the variation is so huge for storms,
#that's why the guess
#Is so off. It's just due to the unpredictability of weather.

rhurr*(hurrstd/yearstd)*(2013-yearmean)+hurrmean
yearsVcounties$hurricanes[grep('2013',years)]
#Predicted value was 386.7426, actual was 18.6666
#So the reason that this one is way off is due, tragically,
#to hurricane Katrina If you look back on the data to 2005,
#you will see that there were 3997 counties affected that
#year by hurricanes. Katrina destroyed much of Louisiana
#and killed many people that year. And so the data point
#is off because of that aweful, infamous outlier.

rflod*(flodstd/yearstd)*(2013-yearmean)+flodmean
yearsVcounties$flooding[grep('2013',years)]
#The predicted was 386.6038, the actual was 287.8333.
#This prediction did so well due to pure luck of the draw.
#Eyeballing it,the variance is very large on with
#flooding, just like storms.

rflod*(tornstd/yearstd)*(2013-yearmean)+tornmean
yearsVcounties$tornado[grep('2013',years)]
#The predicted was 107.1862, the actual was 78.1666
#Once again this was by pure chance, because looking at the
#scatter plot it appears impossible to make a prediction about these
#things. I mean don't get me wrong, it's close, but the previous year
# it was closer to 35, so the predicting power is not that phenominal.
#Especially since these things are not trivial, they're tornadoes.


#However, for the all but the hurricanes prediction,
#they were within 100% error, so they are generally good predictions.

#4.9 D!
```

```
#Contemporary Temperatures list.
contemp_temps <- 1965:2015
for(i in 1965:2015){contemp_temps[i-1964] <- gtemp$Anomaly[gtemp$Year==i]}
plot(contemp_temps,type='l')
#It matches the professor's plot in the book.
#I would like to make something very clear!
#The professor wrote the textbook sometime between 2011 and 2012
#So he was using data that has since changed! So the means and
#standard deviations described as correct in 4.7 are no longer reproducable

meanct <- mean(contemp_temps)
ctstd <- sd(contemp_temps)

#for the record, this following loop, which standardized the data,
#did nothing!
for(i in 1:length(contemp_temps)){contemp_temps[
  i] <- (contemp_temps[i]-meanct)/ctstd}


rStemps<- cor(contemp_temps,yearsVcounties$standardStor)
#The correlation between temps and storms is 0.6415162

rStemps*(storstd/ctstd)*(0.6-meanct)+stormean
rStemps*(storstd/ctstd)*(0.7-meanct)+stormean
#The prediction for the number of counties affected by storms
#when temp=.6 is 648.1323
#The rediction for the number of counties affected by storms
#when temp=.7 is 725.3786


#4.9 D Answers!

rHtemps <- cor(contemp_temps,yearsVcounties$standardHurr)

rHtemps*(hurrstd/ctstd)*(0.6-meanct)+hurrmean
rHtemps*(hurrstd/ctstd)*(0.7-meanct)+hurrmean
#The prediction for the number of counties affected
#by hurricanes when temp=.6 is 442.8398
#The rediction for the number of counties affected
#by hurricanes when temp=.7 is 510.1701

rFtemps <- cor(contemp_temps,yearsVcounties$standardFlod)

rFtemps*(flodstd/ctstd)*(0.6-meanct)+flodmean
rFtemps*(flodstd/ctstd)*(0.7-meanct)+flodmean
#The prediction for the number of counties affected
#by flooding when temp=.6 is 303.1036
#The rediction for the number of counties affected
#by flooding when temp=.7 is 328.1063


rTtemps <- cor(contemp_temps,yearsVcounties$standardTorn)
```

```
rTtemps*(tornstd/ctstd)*(0.6-meanct)+tornmean
rTtemps*(tornstd/ctstd)*(0.7-meanct)+tornmean
#The prediction for the number of counties affected
#by tornadoes when temp=.6 is 121.213
#The rediction for the number of counties affected
#by tornadoes when temp=.7 is 134.3093

#4.9 E!
#The results of 4.9 D show that according to the model, all of the
#types of disasters modeled will affect more counties with an
#increase in global temperature.

#4.9 F!
#The global temperatures trend seems to indicate the earth will continue to
#get warmer(4.9 G). Therefore according to 4.9 D,
#more counties will be affected by
#Disasters in the future. And, because the question tells us to consider
#counties as a general indicator of population, it is true that more people
#in the United States will be affected by disasters in the future.

#4.9 G!

#This plot clearly shows the trend is for the global temperature to increase.
plot(gtemp$Year,gtemp$Anomaly,xlab='Year',
ylab='Absolute Temperature of Earth', main='Year vs Temperature')
abline(lm(Anomaly~Year,data=gtemp))
```

Below is the table that 4.9 A asks for.

"storms"	"hurricanes"	"flooding"	"tornado"	"years"
56	56	213.5	46.5	1965
18.5	0	95.5	1	1966
51.5	29	87.5	18	1967
42.3333333333333	14	85.3333333333333	19.3333333333333	1968
89.3333333333333	34	365.333333333333	15.3333333333333	1969
36.6666666666667	7	153.166666666667	12.1666666666667	1970
23.5	21	55	22.5	1971
357.5	0	152.5	0	1972
273	0	460	125	1973
128.5	10	258.5	187	1974
203	0	195.5	58.5	1975
86.1666666666667	3	86.1666666666667	16.6666666666667	1976
209.5	0	76	3.5	1977
299.5	0	99	33.5	1978
163	105.5	132.5	64	1979
49.6666666666667	12	37.1666666666667	12.1666666666667	1980
20.1666666666667	0	20.1666666666667	3.66666666666667	1981
65.8333333333333	0	56.3333333333333	30.8333333333333	1982
82.8333333333333	9	70.8333333333333	36.3333333333333	1983
94	6	95	58	1984
111	73	104.5	8.5	1985
116.5	0	164	2.5	1986
68.6666666666667	5	63.6666666666667	15.6666666666667	1987
20.5	3	3	17.5	1988
144	113	130	51	1989
227	5	217	106	1990
342.333333333333	34.5	154.833333333333	38.3333333333333	1991
128.833333333333	46	101.833333333333	51.3333333333333	1992
965.666666666667	1	359.166666666667	104.166666666667	1993
340.333333333333	0	196.333333333333	39.3333333333333	1994
176.166666666667	127	221.666666666667	76.1666666666667	1995
647.333333333333	200.5	375.333333333333	14.8333333333333	1996
531.166666666667	0	313.166666666667	67.6666666666667	1997
650.833333333333	266	338.333333333333	128.833333333333	1998
319.5	493	82.5	52	1999
584.5	0	118.5	34	2000
458.333333333333	0	193.333333333333	115.333333333333	2001
464	44	214.5	155.5	2002
469.5	205	272	201.5	2003
716.666666666667	403.833333333333	407.166666666667	133.333333333333	2004
284.5	3997	201.5	3	2005
285.333333333333	0	182.833333333333	39.8333333333333	2006
943	32	323.5	125.5	2007
781.333333333333	566.666666666667	389.333333333333	215.666666666667	2008
763.333333333333	0	306.333333333333	124.333333333333	2009
988.166666666667	64	441.666666666667	118.166666666667	2010
995.833333333333	488	605.833333333333	236.333333333333	2011
475.666666666667	665	54.6666666666667	27.6666666666667	2012
341.333333333333	18.6666666666667	287.833333333333	78.1666666666667	2013
422	0	147	67	2014
389	0	334.5	162.5	2015